型番:ETH-50B　REV.3.00
開発元型式: Ether IO24R

UDP/IP
イーサネット簡単デジタルIOボード

取扱説明書
お使いになる前にこの説明書をよくお読みの上正しくお使いください。
お読みになった後は保管してください。

製品の概要

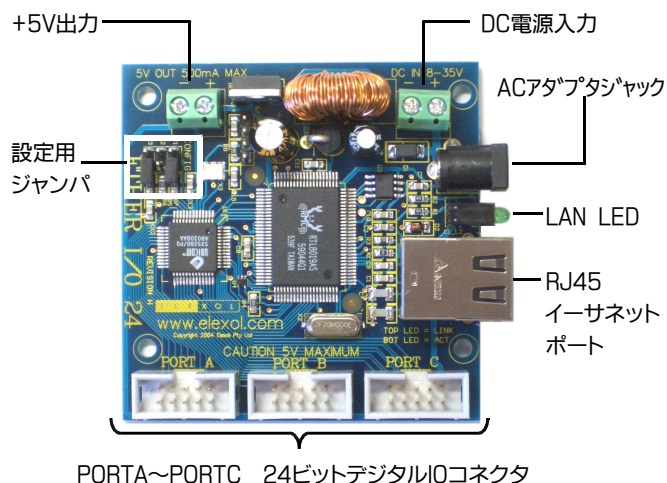
UDP/IP イーサネット簡単デジタルIOボード(型式ETH-50B、以下型式で記載)は、イーサネット経由でUDP/IPプロトコルを用いて、簡単に24ビットのデジタルIOを制御できるボードです。
本体にはイーサネットに直結できるRJ45のコネクタを搭載しており、そのままイーサネットに接続できます。IPアドレスは任意のアドレスを設定できる他、DHCPを搭載していますので、ルーター等によるIPアドレスの自動割付にも対応しています。MACアドレスは書き込み済みですのですぐに使用できます。

デジタルIOはPORTA～PORTCにそれぞれ8ビットずつ、計24ビットのデジタルIOピンを搭載しており、全IOピンは個別に入力又は出力に設定できます。
入力に設定した場合、全ビットに対して個別に、プルアップ設定の有効/無効・スレッショルド電圧をTTLレベルかC-MOSレベルか・シュミットトリガーの有効/無効を設定できます。

ボード内部に不揮発性メモリー(EEPROM)を搭載していますので、各種初期設定値をモジュール本体に登録しておくことができます。

付属品

ETH-50B本体
CD-ROM
マニュアル(本書)

本体の概要

ユーティリティソフトウエアのインストール

ETH-50Bを使用するに際して、各種設定やデモ動作を行うために必要なソフトウエアをパソコンにインストールする必要があります。
各種設定などは、このユーティリティソフトウエアを用いてネットワーク経由で行います。

なお、ETH-50Bにはソフトウエアが3種類付属しています。
1つは、各種設定などをすべて行うことができるユーティリティソフトウエア、2つ目はVisual Basic6.0(VB6)の開発環境で作られた簡単な制御を行うだけのデモプログラム(ソースコード付属)、3つ目はVisual C#2005(C#2005)の開発環境で作られた簡単な制御を行うだけのデモプログラム(ソースコード付属)です。

各種設定などは、ユーティリティソフトウエアで行います。自作アプリケーションを作る場合には開発言語に応じてVB6かC#2005のソースコードを参考にして開発することができます。

1 CD-ROMをパソコンのドライブに挿入して、エクスプローラー等で中身を表示します。

2 "ETH-50Bユーティリティソフトウエア"フォルダを開き、"SETUP.EXE"をダブルクリックして実行します。
→インストーラーが起動します。

3 画面の指示に従ってインストールしてください。

本ユーティリティーソフトウエアは、.NetFramework2.0環境下で動作します。インストーラーには.NetFramework2.0が含まれております。
インストール完了時ダイアログに、".NetFramework2.0をインストールしますか?"というチェックボックスが表示されます。インストールする場合には、チェックを入れて"完了"ボタンを押して下さい。
なお、インストールが必要かどうか分からない場合には、インストールを実行してください。必要がなければ自動的にインストールは中止されます。

電源の接続

電源は、電源入力端子又はACアダプタジャックから給電します。
ACアダプタを使用する場合には、極性がセンタープラスで2.1φのものをご使用ください。出力はDC9V～12Vで500mA以上取り出せるものをご選定ください。

ETH-50B本体の消費電力は、平均1.1Wです。DC9Vの場合本体だけの消費電流は約150mAです。ETH-50Bでは全IOピンから合計最大で210mA(絶対最大定格値)の電流が取り出せますので、少なくとも500mA以上の電流が取り出せる電源をご使用ください。

設定用ジャンパーについて

ETH−50Bの基板上にあるJ1〜J4のジャンパーは、各種設定を行うためのものです。ジャンパーソケットを装着すると下記のような機能が有効になります。

J1　ソケット装着で設定記憶用EEPROMのロック
J2　ソケット装着でDHCPが有効になりEEPROMに設定されているIPアドレスは無視される、ソケットなしでEEPROMに設定されたIPアドレスがETH−50Bに設定される
J3　ソケット装着でIPアドレスを10.10.10.10に固定
J4　ソケット装着でIPアドレスを192.168.0.10に固定

※工場出荷時は、J2にソケットが装着されており、DHCP機能が有効になっています。

搭載プロトコルと使用ポート

ETH−50Bは、イーサネットでの運用、データ伝送をするために必要な各種プロトコルを実装しています。
実装しているプロトコルは、ARP・BOOTP・DHCP・ICMP・UDP/IPです。

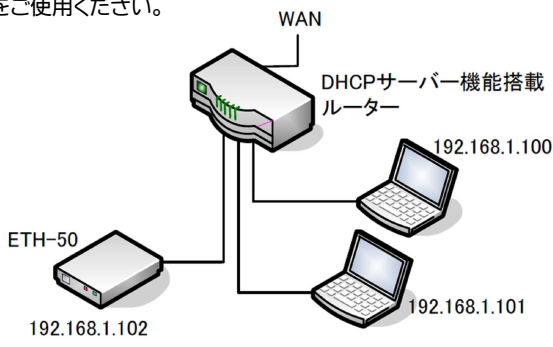
ETH−50Bはデータ伝送にUDP/IPプロトコルを使用します。
使用ポートは、2424です。受信側、送信側ともにポート2424を使用します。

※ポート番号は固定されておりユーザーが変更することはできません。

イーサネットへの接続とIPアドレスの割り振りについて

ネットワークに接続された機器同士が通信するには、IPアドレスが正しく割り振られている必要があります。
ETH−50Bは、DHCPを搭載していますので、DHCPサーバー機能のあるルーターやハブがネットワーク上にある場合、ETH−50Bを接続するだけで使用できます。

ネットワークの環境に応じて設定方法が異なりますので下記の手順に従い設定を行ってください。
なお、設定に際してはジャンパーソケットJ1〜J4を使用します。

■DHCPサーバーのあるネットワークへの接続
ETH−50Bでは、DHCP機能を搭載していますので、接続するネットワーク上にDHCPサーバーがあれば、動的にIPアドレスが割り当てられ、そのまま使用できるようになります。
工場出荷時の設定ではDHCP機能は有効になっています
ルーターもしくはハブと接続する場合には、全結線のストレートケーブルをご使用ください。

接続後にETH−50Bの電源を投入すると、DHCPにより自動的にIPアドレスが割り振られて使用できるようになります。

ETH−50B本体にある設定用ジャンパーのJ2にソケットを装着すると、DHCP機能が有効になります。（工場出荷時の状態です）

※J2のジャンパーをショートすると、DHCPが有効となります。本体のEEPROMにIPアドレスが登録してある場合でもDHCPの設定が優先されます。IPアドレスを固定にしたい場合には、ユーティリティーソフトウエアにてIPアドレスを設定して本体のEEPROMに書き込む必要があります。

■DHCPサーバーがないネットワークへの接続
DHCPがない場合、ETH−50BのIPアドレスは手動で設定することになります。DHCPサーバーがないネットワーク上でのIPアドレスの手動設定手順は下記の通りです。なお、DHCPがあるネットワーク上で、固定IPアドレスを設定したい場合には手順が異なりますので、この方法では設定しないでください。

1　ETH−50Bの電源を切断し、本体にある設定用ジャンパーのJ3又はJ4をショートし、ETH−50Bの電源を投入します。J3をショートするか、J4をショートするかで起動時の本体のIPアドレスが異なります。どちらかを選んで下さい。

J3をショートした場合・・・・　10.10.10.10
J4をショートした場合・・・・　192.168.0.10

※J1、J2のジャンパーは必ず外しておいて下さい。
※J3とJ4を同時に両方ショートさせないで下さい。

2　J3又はJ4をショートして電源を投入すると、DHCP機能は無効となり、本体のIPアドレスは強制的に設定されます。

3　パソコンとETH−50BをクロスLANケーブルで接続するか、ルーターを介してネットワークに接続します。
パソコンと同じネットワークにETH−50Bがないと設定ができませんので、パソコンのIPアドレスを同じネットワークアドレスになるように設定します。

例えば、J3をショートしている場合にはパソコンのIPアドレスを下記のように設定します。

IPアドレス　　　10.10.10.11
サブネットマスク　　255.255.255.0

同じように、J4をショートしている場合には、下記のようにパソコンのアドレスを設定します。

IPアドレス　　　192.168.0.11
サブネットマスク　　255.255.255.0

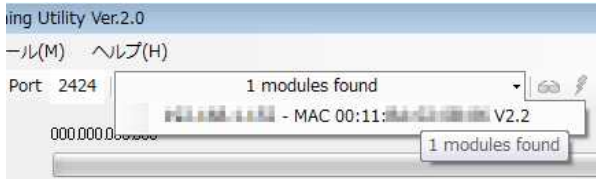
※上記例は参考例です。ネットワークアドレス部が同じであれば、ホストアドレス部は異なるアドレスでもかまいません。

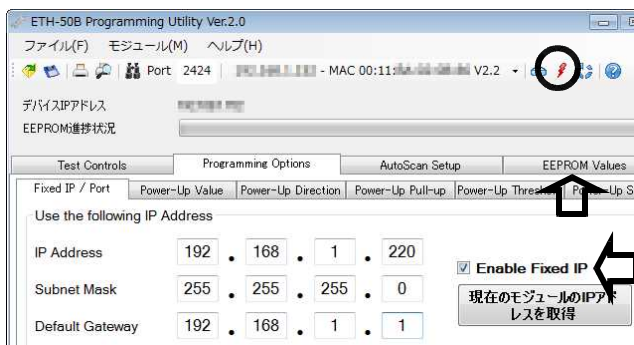
※パソコンのIPアドレスとサブネットマスクの設定方法はパソコンの取扱マニュアル等をご覧下さい。

4 ETH-50Bのユーティリティソフトウエアを起動します。
※スタート→プログラム→Microtechnica→ETH-50B Utility2.00→ETH-50B Programming Utility Softwareの順で起動します。

5 下記の検索ボタンを押すと、接続されたETH-50Bが検出されます。

検索ボタン

ETH-50Bが見つかると"1 modules found"と表示されますので、クリックして一覧を表示させます。

デバイスをクリックするとそのデバイスが選択されます。

6 "Programming Options"タブをクリックして開き、その下にある"Fixed IP/Port"タブを開きます。"Enable Fixed IP"のチェックボックスにチェックを入れて、その下にあるボックスに、ETH-50Bに設定したいIPアドレスとサブネットマスク、ETH-50Bが接続されているネットワークにあるデフォルトゲートウェイのアドレスを入力します。

入力が完了したら、"EEPROM Values"タブに移動します。
"EEPROMに書き込み"ボタンを押すと書き込みが実行されます。
プログレスバーが表示されます。100%になると"EEPROM Write Successful!"と表示されますので、OKボタンを押して閉じます。

7 ETH-50B本体のJ3又はJ4ジャンパーソケットを外します。
ETH-50Bの電源を一度切断して、再度投入します。
これでIPアドレスが書き換わりました。
書き換わった後は、新しいIPアドレスでパソコンからアクセスしますので、パソコンのIPアドレスとETH-50BのIPアドレスは互いに同じサブネットに属するように、パソコン側のIPアドレス設定を変更してください。

設定したIPアドレスを有効にするためには、J2～J4のジャンパーソケットは外しておく(オープンにしておく)必要があります。

ファイヤーウォールの設定について

パソコンにセキュリティーソフトウエアやウイルス対策ソフトウエアがインストールされている場合ファイヤーウォールが設定されていることがあります。WindowsXP SP2やWindowsVistaにもファイヤーウォールが標準で装備されています。

これらのファイヤーウォールがある場合、ETH-50Bとパソコンとの通信がファイヤーウォールによって遮断されてしまうことがあります。
ご使用に際しては必ず、ポート2424はETH-50Bとの通信用としてファイヤーウォールを解放してください。
ファイヤーウォールの操作方法は、セキュリティーソフトウエアのマニュアルをご覧ください。

■IPアドレスの確認とpingでの確認

IPアドレスが正しく設定され、パソコンと送信ができるかどうかを確認します。

1 ユーティリティソフトウエアを起動します。
モジュール検索ボタンを押します。右側のウインドウにIPアドレスが表示されETH-50B本体のMACアドレスが表示されれば成功です。

表示を確認します

※もし、正しくETH-50Bが検出できない場合にはETH-50BのIPアドレスが正しく設定できていないか、パソコンとETH-50Bが同一サブネット上に存在していないことが考えられます。今一度、パソコンのIPアドレスと、ETH-50BのIPアドレスをご確認ください。
また、パソコンにウイルス対策ソフトやセキュリティーソフトウエアがインストールされている場合、それらのソフトウエアのファイヤーウォール機能が有効になっていて、パケットが遮断されていることがあります。ファイヤーウォールの設定において、ポート2424が解放されていることをご確認ください。

2 上記の手順1で正しくIPアドレスとMACアドレスが表示されれば、通信は確立されていますが、ETH-50BはICMPを搭載していますので、通信をpingコマンドで確認することも可能です。

```
C:¥>ping 192.168.1.190

Pinging 192.168.1.190 with 32 bytes of data:

Reply from 192.168.1.190: bytes=32 time=35ms TTL=128
Reply from 192.168.1.190: bytes=32 time=2ms TTL=128
Reply from 192.168.1.190: bytes=32 time=2ms TTL=128
Reply from 192.168.1.190: bytes=32 time=2ms TTL=128

Ping statistics for 192.168.1.190:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 35ms, Average = 10ms
```

Lost=0となっていればOKです。

※ジャンパー設定についての注意
J2～J4のジャンパー設定は、ETH－50BのIPアドレスをどのように定めるかということを設定するためのものです。

J2は、DHCP機能を有効にするか、DHCP機能を無効にして本体のEEPROMに設定されている静的なIPアドレスにするかということを決めるものです。J2をショートするとDHCP機能が有効となり、ETH－50BのIPアドレスはDHCPサーバーから割り振られたものとなり、EEPROMに書き込まれているIPアドレスは無視されます。
J2をオープンにすると、DHCP機能は無効となり、EEPROMに設定されたIPアドレスが有効となります。

J3とJ4はDHCPサーバーが存在しないネットワークでETH－50Bを使用する際にIPアドレスを手動設定するときに使用するものです。本体のIPアドレスを強制的に10.10.10.10又は192.168.0.10に設定できます。ETH－50ユーティリティーソフトウエアを使用することで、IPアドレスの設定ができます。設定したIPアドレスは再起動後に有効となり、J2ピンがオープンである必要があります。J2がショートされている場合にはDHCP機能が有効となっており、EEPROMに書き込まれたIPアドレスは無視されてしまいます。

デジタルIOピンについて

デジタルIOピンは、PORTA～PORTCに各8ビットずつあり、計24ビットのピンがあります。各ピンは、個別に入力か出力かに設定ができます。

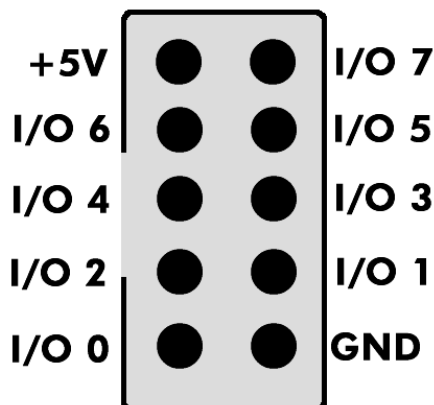
■出力電圧と出力できる電流について
出力に設定した場合、出力電圧は5Vp－pのTTLレベルとなります。
各ビットからは最大で30mAの電流が取り出せます。
また全24ビット合計で最大200mAの電流を取り出すことができます。
これらの値を超えると本体が故障しますので、この範囲を絶対に超えないよう周辺回路の設計をして頂けますようお願いいたします。

誘導性負荷（例えば電磁式リレーやモーター）などは、本体の故障の原因となりますので、ETH－50BのIOピンから直接駆動しないでください。

■入力電圧と入力できる電流について
入力に設定した場合、各ピンのスレッショルド電圧（閾値電圧）はTTLスレッショルドかC－MOSスレッショルドかを選択できます。
また、シュミットトリガーの有効・無効も個別に設定可能です。各電圧値は下表の通りとなります。

入力論理	設定	最小値	最大値
Lowレベル	TTL設定	0V	0.8V
	C－MOS設定	0V	1.5V
	ST設定	0V	0.75V
Highレベル	TTL設定	2.0V	5.0V
	C－MOS設定	3.5V	5.0V
	ST設定	4.25V	5.0V

入力設定にした場合には、各ピンはハイインピーダンスとなりますが、ピンをHighレベルのした時に、最大で約3.0μA程度の電流が流れることがあります（5V時）。

■ピン配列
各PORTのピン配置は下記の通りです。

UDPコマンドの概要

ETH-50Bは、UDPコマンドにて各種操作をします。本項では、基本的なUDPコマンドのフォーマットと使用方法を解説しています。なお、16進数の場合には数値の前に0xを付けて表現しています。

■ETH-50Bのコマンドフォーマット

ETH-50Bのコマンドは基本的に1バイト単位で、ASCIIコードで送信します。コマンドの詳細は後述しますが、ここでは下記のような例を取り上げます。

例えば、PORTAの出力を操作したい場合、コマンドは'A'です。その後にPORTAの値を送信します。出力の場合0がLow、1がHighレベルとなります。PORTAのビット1だけをHに、残りをLにしたい場合には00000001となりますので、16進数では0x01です。'A'は0x41ですので、送信するデータは2バイトで"0x41 0x01"をUDPで送信することになります。

戻り値も同様で、ポート名の後に値が返ります。例えば、PORTAの状態を取得する場合には、'a'コマンドを使用します。'a'は、0x61ですので、UDPで0x61を送信すると、直ちに "'A' *data*" の2バイトが返ります。

主に使用するコマンドを下記に紹介します。

機能	コマンド	戻り値
PORTAに値を出力	A *data*	
PORTBに値を出力	B *data*	
PORTCに値を出力	C *data*	
PORTAの状態を取得	a	A *data*
PORTBの状態を取得	b	B *data*
PORTCの状態を取得	c	C *data*

機能	コマンド	戻り値
PORTAの入出力方向の設定	!A *data*	
PORTBの入出力方向の設定	!B *data*	
PORTCの入出力方向の設定	!C *data*	

■ホストソフトウエアの作成について

パソコン側のホストソフトウエアは、UDP通信ができるようにプログラムを作ります。ETH-50Bでは、付属のCD-ROMにVisual Basic6.0(VB6)のサンプルプログラムと、Visual C#2005(VC#2005)のサンプルプログラムをソースファイルごと収録しています。

VB6で開発をする場合には、UDP通信が簡単に行えるActiveXコントロールのWinsockを使用します。受信、送信ともに簡単にプログラミングが行えます。Winsockコントロールはコンポーネントの"Microsoft Winsock Control6.0"として追加できます。

Winsockコントロールを、Formに貼り付けて使用します。
使い方や書式は、サンプルプログラムを参考にしてください。

VC#2005、VB2005などの.Net Frameworkを使用する開発言語で開発する場合には、System.Net.Sockets名前空間に属するUdpClientクラスを使用すると簡単に通信が行えます。
Visual Basic2005なども言語体系は違いますが基本的には同じですので、VC#2005のサンプルプログラムを参考にできます。
下記は、UdpClientクラスでデータを送信・受信するプログラムの例です。ポートは2424に設定します。

```
System.Text.Encoding enc = System.Text.Encoding.UTF8;

  string remoteHost = "192.168.1.100";

  System.Net.Sockets.UdpClient udp =
    new System.Net.Sockets.UdpClient(2424);

    byte[] SendData = new byte[4];
    SendData[0] = 0x49;
    SendData[1] = 0x4F;
    SendData[2] = 0x32;
    SendData[3] = 0x34;

    udp.Send(SendData, 4, remoteHost, 2424);

    IPEndPoint remoteEP = null;
    byte[] recBytes = udp.Receive(ref remoteEP);
    string recData = enc.GetString(recBytes);

    recBytes = udp.Receive(ref remoteEP);

    udp.Close();
```

送信するデータはnew演算子でbyte型配列を作り、そこにデータを代入します。送信は、UdpClientクラスのSendメソッドで送信バイト数を指定して送信できます。
受信は、Endpoint(IPアドレスと、ポート番号を組み合わせたもの)を指定してUdpClientクラスのReceiveメソッドで行います。

実際の使用方法については、付属のサンプルプログラムの内容を参考にしてください。

※サンプルプログラムは、CD-ROMの"サンプルプログラム"フォルダに収録されています。

UDPコマンド一覧

実行される内容	PORTAの入出力方向を設定する
コマンド(16進数)	!A　(0x21 0x41)　※2バイト
引数	0x00～0xFF 入出力方向の値(入力=1、出力=0)
戻り値	なし
詳細	PORTAの各ビットの入出力方向を設定します。該当のビットが0で出力に、1で入力に設定されます。 入力に設定すると、ハイインピーダンスになります。出力に設定した場合、1つのピンから取り出せる電流は最大30mAまで、全24ビットから取り出せる合計の電流は200mAまでです。
例	全ビット出力に設定する場合 0x21 0x41 0x00

実行される内容	PORTBの入出力方向を設定する
コマンド(16進数)	!B　(0x21 0x42)　※2バイト
引数	0x00～0xFF 入出力方向の値(入力=1、出力=0)
戻り値	なし
詳細	PORTBの各ビットの入出力方向を設定します。該当のビットが0で出力に、1で入力に設定されます。 入力に設定すると、ハイインピーダンスになります。出力に設定した場合、1つのピンから取り出せる電流は最大30mAまで、全24ビットから取り出せる合計の電流は200mAまでです。
例	全ビット入力に設定する場合 0x21 0x42 0xFF

実行される内容	PORTCの入出力方向を設定する
コマンド(16進数)	!C　(0x21 0x43)　※2バイト
引数	0x00～0xFF 入出力方向の値(入力=1、出力=0)
戻り値	なし
詳細	PORTCの各ビットの入出力方向を設定します。該当のビットが0で出力に、1で入力に設定されます。 入力に設定すると、ハイインピーダンスになります。出力に設定した場合、1つのピンから取り出せる電流は最大30mAまで、全24ビットから取り出せる合計の電流は200mAまでです。
例	ビット1とビット5を入力に、その他を出力に設定する場合 0x21 0x43 0x22

実行される内容	PORTAの出力を制御する
コマンド(16進数)	A　(0x41)　※1バイト
引数	0x00～0xFF 出力する値(H=1、L=0)
戻り値	なし
詳細	PORTAの各ビットに値を出力します。 該当ビットが1でHレベル、0でLレベルとなります。 Hレベルの時、1つのピンから取り出せる電流は最大30mAまで、全24ビットから取り出せる合計の電流は200mAまでです。 Lレベルの時、1つのピンに流し込める電流は最大25mAまでです。
例	PORTAのビット5だけをHにする場合 0x41 0x20

実行される内容	PORTBの出力を制御する
コマンド(16進数)	B　(0x42)　※1バイト
引数	0x00～0xFF 出力する値(H=1、L=0)
戻り値	なし
詳細	PORTBの各ビットに値を出力します。 該当ビットが1でHレベル、0でLレベルとなります。 Hレベルの時、1つのピンから取り出せる電流は最大30mAまで、全24ビットから取り出せる合計の電流は200mAまでです。 Lレベルの時、1つのピンに流し込める電流は最大25mAまでです。
例	PORTBのビット2と7だけをHにする場合 0x42 0x84

実行される内容	PORTCの出力を制御する
コマンド(16進数)	C　(0x43)　※1バイト
引数	0x00～0xFF 出力する値(H=1、L=0)
戻り値	なし
詳細例	PORTCの各ビットに値を出力します。 該当ビットが1でHレベル、0でLレベルとなります。 Hレベルの時、1つのピンから取り出せる電流は最大30mAまで、全24ビットから取り出せる合計の電流は200mAまでです。 Lレベルの時、1つのピンに流し込める電流は最大25mAまでです。
	PORTCの全ビットをHにする場合 0x43　0xFF

実行される内容	PORTAの内蔵プルアップを設定する
コマンド(16進数)	**@A**　(0x40 0x41)　※2バイト
引数	0x00～0xFF 設定する値(プルアップ有効=0、無効=1)
戻り値	なし
詳細	PORTAの内蔵プルアップを設定します。プルアップ設定は、該当ピンが出力に設定されている場合には関係ありません。入力に設定されている時だけ、その設定が関係します。 該当ビットを0にするとプルアップが有効に、1にすると無効になります。プルアップを有効にすると、入力設定時、定常状態がHレベル(1)になります。Lレベルの印加で0になります。 電源投入時は、プルアップが全ピン無効(0xFF)に設定されています。
例	PORTAのビット0～ビット4のプルアップを有効にする 0x40 0x41 0xE0

実行される内容	PORTBの内蔵プルアップを設定する
コマンド(16進数)	**@B**　(0x40 0x42)　※2バイト
引数	0x00～0xFF 設定する値(プルアップ有効=0、無効=1)
戻り値	なし
詳細	PORTBの内蔵プルアップを設定します。プルアップ設定は、該当ピンが出力に設定されている場合には関係ありません。入力に設定されている時だけ、その設定が関係します。 該当ビットを0にするとプルアップが有効に、1にすると無効になります。プルアップを有効にすると、入力設定時、定常状態がHレベル(1)になります。Lレベルの印加で0になります。 電源投入時は、プルアップが全ピン無効(0xFF)に設定されています。
例	PORTBの全ビットのプルアップを有効にする 0x40 0x41 0x00

実行される内容	PORTCの内蔵プルアップを設定する
コマンド(16進数)	**@C**　(0x40 0x43)　※2バイト
引数	0x00～0xFF 設定する値(プルアップ有効=0、無効=1)
戻り値	なし
詳細	PORTCの内蔵プルアップを設定します。プルアップ設定は、該当ピンが出力に設定されている場合には関係ありません。入力に設定されている時だけ、その設定が関係します。 該当ビットを0にするとプルアップが有効に、1にすると無効になります。プルアップを有効にすると、入力設定時、定常状態がHレベル(1)になります。Lレベルの印加で0になります。 電源投入時は、プルアップが全ピン無効(0xFF)に設定されています。
例	PORTCの全ビットのプルアップを無効にする 0x40 0x41 0xFF

実行される内容	PORTAのスレッショルド電圧を設定する
コマンド(16進数)	**#A**　(0x23 0x41)　※2バイト
引数	0x00～0xFF 設定する値
戻り値	なし
詳細	PORTAのスレッショルド電圧(閾値電圧)を設定します。入力電圧の閾値は該当ビットが0の時、2.5V、1の時1.4Vになります。 該当ビットが出力に設定されている場合、この設定は関係ありません。 電源投入時は、全ピン1.4V設定(0xFF)になっています。
例	PORTAの全ビットをTTLスレッショルドにする 0x23 0x41 0xFF

実行される内容	PORTBのスレッショルド電圧を設定する
コマンド(16進数)	**#B**　(0x23 0x42)　※2バイト
引数	0x00～0xFF 設定する値
戻り値	なし
詳細	PORTBのスレッショルド電圧(閾値電圧)を設定します。入力電圧の閾値は該当ビットが0の時、2.5V、1の時1.4Vになります。 該当ビットが出力に設定されている場合、この設定は関係ありません。 電源投入時は、全ピン1.4V設定(0xFF)になっています。
例	PORTBの全ビットをC-MOSｽﾚｯｼｮﾙﾄﾞにする 0x23 0x42 0x00

実行される内容	PORTCのスレッショルド電圧を設定する
コマンド(16進数)	**#C**　(0x23 0x43)　※2バイト
引数	0x00～0xFF 設定する値
戻り値	なし
詳細	PORTCのスレッショルド電圧(閾値電圧)を設定します。入力電圧の閾値は該当ビットが0の時、2.5V、1の時1.4Vになります。 該当ビットが出力に設定されている場合、この設定は関係ありません。 電源投入時は、全ピン1.4V設定(0xFF)になっています。
例	PORTCの全ビットをC-MOSｽﾚｯｼｮﾙﾄﾞにする 0x23 0x43 0x00

実行される内容	PORTAのシュミットトリガーを設定する
コマンド(16進数)	**$A** (0x24 0x41) ※2バイト
引数	0x00～0xFF 設定する値
戻り値	なし
	PORTAのシュミットトリガーを設定します。 該当ビットが0の時シュミットトリガーが有効に、1の時ノーマル入力になります。 該当ビットが出力に設定されている場合、この設定は関係ありません。 電源投入時は、全ピンノーマル入力設定(0xFF)になっています。
例	PORTA全ビットのｼｭﾐｯﾄﾄﾘｶﾞｰを有効にする 0x24 0x41 0x00

実行される内容	PORTBのシュミットトリガーを設定する
コマンド(16進数)	**$B** (0x24 0x42) ※2バイト
引数	0x00～0xFF 設定する値
戻り値	なし
詳細	PORTBのシュミットトリガーを設定します。 該当ビットが0の時シュミットトリガーが有効に、1の時ノーマル入力になります。 該当ビットが出力に設定されている場合、この設定は関係ありません。 電源投入時は、全ピンノーマル入力設定(0xFF)になっています。
例	PORTB全ビットのｼｭﾐｯﾄﾄﾘｶﾞｰを無効にする 0x24 0x42 0xFF

実行される内容	PORTCのシュミットトリガーを設定する
コマンド(16進数)	**$C** (0x24 0x43) ※2バイト
引数	0x00～0xFF 設定する値
戻り値	なし
詳細	PORTCのシュミットトリガーを設定します。 該当ビットが0の時シュミットトリガーが有効に、1の時ノーマル入力になります。 該当ビットが出力に設定されている場合、この設定は関係ありません。 電源投入時は、全ピンノーマル入力設定(0xFF)になっています。
例	PORTC全ビットのｼｭﾐｯﾄﾄﾘｶﾞｰを無効にする 0x24 0x43 0xFF

実行される内容	PORTAの状態を取得する
コマンド(16進数)	**a** (0x61) ※1バイト
引数	なし
戻り値	'A'に続き、ポートの状態(8ビット) ※2バイト
詳細	PORTAの状態を取得します。 状態は、各ピンがHの時は1、Lの時は0となり0x00～0xFFまでの8ビットで返されます。 主にポートが入力方向に設定されている時に使用するコマンドですが、出力方向に設定されている場合には現在の出力されている値が取得できます。
例	PORTAのピンが、HHHHLLLLであった場合 0x61 （送信） 0x41 0xF0 （受信）

実行される内容	PORTBの状態を取得する
コマンド(16進数)	**b** (0x62) ※1バイト
引数	なし
戻り値	'B'に続き、ポートの状態(8ビット) ※2バイト
詳細	PORTBの状態を取得します。 状態は、各ピンがHの時は1、Lの時は0となり0x00～0xFFまでの8ビットで返されます。 主にポートが入力方向に設定されている時に使用するコマンドですが、出力方向に設定されている場合には現在の出力されている値が取得できます。
例	PORTBのピンが、HLHLHLHLであった場合 0x62 （送信） 0x42 0xAA （受信）

実行される内容	PORTCの状態を取得する
コマンド(16進数)	**c** (0x63) ※1バイト
引数	なし
戻り値	'C'に続き、ポートの状態(8ビット) ※2バイト
詳細	PORTCの状態を取得します。 状態は、各ピンがHの時は1、Lの時は0となり0x00～0xFFまでの8ビットで返されます。 主にポートが入力方向に設定されている時に使用するコマンドですが、出力方向に設定されている場合には現在の出力されている値が取得できます。
例	PORTCのピンが、LLLLLLLLであった場合 0x63 （送信） 0x43 0x00 （受信）

実行される内容	PORTAの入出力方向を取得する
コマンド(16進数)	**!a** (0x21 0x61)　※2バイト
引数	なし
戻り値	'!A'に続き、ポートの状態(8ビット)　※3バイト
詳細	PORTAの入出力方向の値を取得します。 各ピンが入力の場合は1、出力の場合は0となり 0x00～0xFFまでの8ビットで返されます。
例	PORTAが、全ピン入力の場合 0x21 0x61 (送信) 0x21 0x41 0xFF (受信)

実行される内容	PORTBの入出力方向を取得する
コマンド(16進数)	**!b** (0x21 0x62)　※2バイト
引数	なし
戻り値	'!B'に続き、ポートの状態(8ビット)　※3バイト
詳細	PORTBの入出力方向の値を取得します。 各ピンが入力の場合は1、出力の場合は0となり 0x00～0xFFまでの8ビットで返されます。
例	PORTBが、全ピン出力の場合 0x21 0x62 (送信) 0x21 0x42 0x00 (受信)

実行される内容	PORTCの入出力方向を取得する
コマンド(16進数)	**!c** (0x21 0x63)　※2バイト
引数	なし
戻り値	'!C'に続き、ポートの状態(8ビット)　※3バイト
詳細	PORTCの入出力方向の値を取得します。 各ピンが入力の場合は1、出力の場合は0となり 0x00～0xFFまでの8ビットで返されます。
例	PORTCが、全ピン入力の場合 0x21 0x63 (送信) 0x21 0x43 0xFF (受信)

実行される内容	ETH-50BのMACアドレスとファームウエアバージョンを取得する
コマンド(16進数)	**IO24** (0x49 0x4F 0x32 0x34) ※4バイト
引数	なし
戻り値	"IO24"の文字列に続き、6バイトのMACアドレス 2バイトのファームウエアバージョン ※全12バイト
詳細	ETH-50B本体のMACアドレスと、ファームウエアバージョンが返ります。 データの先頭には"IO24"の文字列が4バイト、それに続き、6バイトのMACアドレス、最後に2バイトのファームウエアバージョンが返ります。 このコマンドをブロードキャストで送信することでネットワーク上のモジュールを検出することができます。送られたパケットを解析することで、モジュールのIPアドレスを取得できます。 ※詳しくは、「ネットワーク上のETH-50B検出方法」の項をご覧ください。

ネットワーク上のETH-50B検出方法

ETH-50BのIPアドレス割り振りがDHCP設定の場合、ETH-50BのIPアドレスはルーターなどによって自動的に割り振られます。
その場合、ネットワーク上のETH-50Bを検出するにはブロードキャストアドレスによってコマンドを送信し、戻り値からETH-50BのIPアドレスを知ることができます。

ブロードキャストアドレスを知るには、サブネットマスクの値が必要です。サブネットマスクの値より、IPアドレスのネットワークアドレス部を指定して、ホストアドレス部をブロードキャストとします。

ブロードキャストアドレスに、コマンド"IO24"(0x49 0x4F 0x32 0x34)を送信して、ETH-50Bが受信すると"IO24"の文字列に続き、6バイトのMACアドレスと、2バイトのファームウエアバージョンの計12バイトを返します。この時、リモートホストのIPアドレスを取得すればETH-50BのIPアドレスを特定できます。

下記にVB6とVC#2005でのプログラム例を紹介します。
いずれも"IO24"をブロードキャストで送信し、"IO24"を返してきたデバイス、すなわちETH-50BのIPアドレスを取得します。

■VB6でETH-50Bを探す場合

```
Private Sub Command1_Click()

Winsock1.RemoteHost = "255.255.255.255"
Winsock1.RemotePort = 2424
Winsock1.Bind

Winsock1.SendData "IO24"

While Winsock1.BytesReceived = 0
Wend

N = Winsock1.BytesReceived
Winsock1.GetData a$, vbString, N

MsgBox (Winsock1.RemoteHostIP)

End Sub
```

※Winsock1.Bindは1回実行した場合、2回目からは必要ありません。
2回実行するとエラーが発生します。

■VC#2005でETH-50Bを探す場合

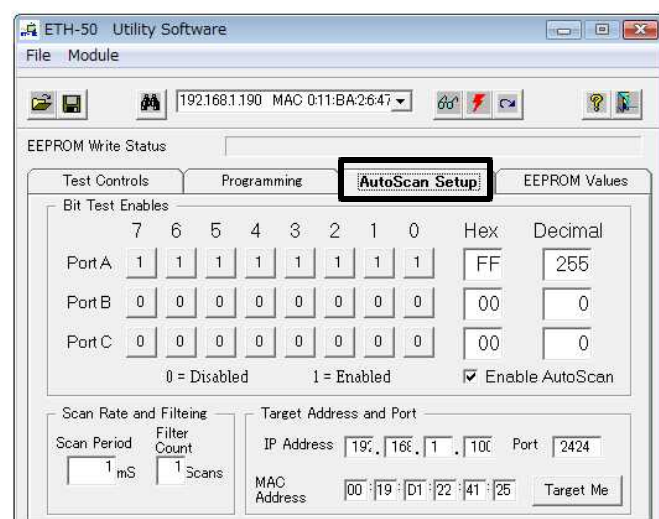
```
private void button1_Click(object sender, EventArgs e)  {

   System.Text.Encoding enc = System.Text.Encoding.UTF8;

   string remoteHost = "255.255.255.255";

   System.Net.Sockets.UdpClient udp =
         new System.Net.Sockets.UdpClient(2424);

   byte[] SendData = new byte[4];
   SendData[0] = 0x49;
   SendData[1] = 0x4F;
   SendData[2] = 0x32;
   SendData[3] = 0x34;

   udp.Send(SendData, 4, remoteHost, 2424);

   IPEndPoint remoteEP = null;
   byte[] recBytes = udp.Receive(ref remoteEP);
   string recData = enc.GetString(recBytes);
   recBytes = udp.Receive(ref remoteEP);

   MessageBox.Show(remoteEP.Address.ToString());
```

ブロードキャストで送信した場合、送信した自らのPC(localhost)も"IO24"を受信するため、1回の受信ではなく上記例では2回受信を行い、2回目に受信したデータの送信元アドレスをETH-50BのIPアドレスとして参照しています。

オートスキャンモード

ETH-50Bは、ポートの状態を定期的に自動スキャンして、変化があった際に自動的に指定したIPアドレスとMACアドレス先にデータを送信するオートスキャンモードを搭載しています。オートスキャンモードを使用することで、ホストコンピューター側がETH-50Bの状態を定期的にポーリングスキャンしなくてもピンに変化があると設定したIPアドレス・MACアドレスのデバイスに対して自動的にポートの値を返してきます。

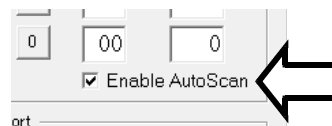
スキャンするピン、スキャンレート(スキャンするタイミング)、ポートのデータを送信する送信先ホストのアドレスはETH-50Bのユーティリティソフトウエアから設定できます。また設定した値は、ETH-50B内部のEEPROM(不揮発性メモリー)に記憶されますので、電源を切断してもその内容は保持されます。

■ユーティリティソフトウエアで設定する
ETH-50Bがネットワークに接続され、ETH-50Bユーティリティソフトウエアで認識できる状態にしておきます。
ETH-50Bを起動して"AutoScan Setup"タブを開きます

1 "Bit Test Enables"の項目でオートスキャンを実行したいピンを選択します。該当ピンのボタンが"0"で無効、"1"で有効になります。
※ETH-50Bは、ここで"1"に指定されたピンを定期的に監視します。

2 スキャンピリオドとフィルターカウントを設定します。
スキャンピリオドはスキャンする間隔をミリ秒単位で指定します。
フィルターカウントは、上記で設定したスキャンピリオドあたり、何回スキャンするかを設定します。1秒あたり1000回(1スキャンが1ミリ秒)が基本となっています。よって、1秒あたり500回のスキャン(1スキャン2ミリ秒)の場合には、1000÷500で"2"と入力します。入力する値は1000から割った値です。この値を増やすと1サンプルのスキャン間隔が長くなりますので、機械スイッチを接続した時のチャタリングやノイズを防止することができます。

一般的には、スキャンピリオドは2000、フィルターカウントを2程度に設定します。ノイズが多い環境などでは、フィルターカウントの値を大きくします。

3 ターゲットアドレスとポート番号を設定します。
ETH-50Bはここで設定したIPアドレス・MACアドレスのデバイスに対してポートの状態を送信してきます。情報を受信したいデバイスのアドレスを設定してください。ポート番号は2424に設定します。

4 オートスキャンモードを有効にするため、Enable AutoScanのチェックボックスにチェックを入れます。

5 設定内容を、EEPROMに書き込みます。
EEPROM Valuesタブに切り替えます。"Write EEPROM"ボタンを押すと設定内容が書き込まれます。書き込み中はプログレスバーが進捗状況を表示します。
なお、書き込み途中でエラーが発生する場合には、下記をおためしください。

①Scan Periodの値とFilter Countの値が正しいか確認ください。

②書き込み前に"Erase EEPROM"をクリックしてEEPROMを消去してから書き込みを実行してください。

③何回か書き込みを実行してみてください。ネットワーク環境によっては正しくデータが届いていない場合があります。

6 書き込みが完了したらモジュールリセットボタンを押して、本体をリセットするか、電源を再起動してください。

"Bit Test Enables"の部分で有効にしたピンの状態が変化すると、ETH-50Bは手順3で設定したデバイスに対して下記の形式でデータを送信してきます。
「0x2A ポート名 ポートの値」
例えばPORTAが全ビットHになった場合には下記のようになります
「0x2A 0x41 0xFF」

状態に変化がない場合ETH-50Bはデータを送信しません。

内蔵EEPROMの扱い方

ETH−50Bは、様々な設定状態などを記録するためにEEPROM(不揮発性メモリー)を搭載しています。通常EEPROMの内容は、付属のユーティリティソフトウエアによって行います。

■メモリーマップ

Word	MSB Byte	Function	LSB Byte	Function
0-4	1-9	**Reserved (Unwritable)**	0-8	**Reserved (Unwritable)**
5	11	Control Bits 2	10	Control Bits 1
6	13	Fixed IP Address Byte 2	12	Fixed IP Address Byte 1
7	15	Fixed IP Address Byte 4	14	Fixed IP Address Byte 3
8	17	Preset Port A Value	16	Preset Port A Direction
9	19	Preset Port A Pull up	18	Preset Port A Threshold
10	21	Preset Port B Direction	20	Preset Port A Schmitt Trigger
11	23	Preset Port B Threshold	22	Preset Port B Value
12	25	Preset Port B Schmitt Trigger	24	Preset Port B Pull up
13	27	Preset Port C Value	26	Preset Port C Direction
14	29	Preset Port C Pull up	28	Preset Port C Threshold
15	31	Reserved for Future Use	30	Preset Port C Schmitt Trigger
16	33	AutoScan Port B Mask	32	AutoScan Port A Mask
17	35	AutoScan Filter Count	34	AutoScan Port C Mask
18	37	AutoScan Scan Rate MSB	36	AutoScan Scan Rate LSB
19	39	AutoScan Target MAC Address 2	38	AutoScan Target MAC Address 1
20	41	AutoScan Target MAC Address 4	40	AutoScan Target MAC Address 3
21	43	AutoScan Target MAC Address 6	42	AutoScan Target MAC Address 5
22	45	AutoScan Target IP Address 2	44	AutoScan Target IP Address 1
23	47	AutoScan Target IP Address 4	46	AutoScan Target IP Address 3
24	49	AutoScan Target Port MSB	48	AutoScan Target Port LSB

内部のメモリーは上記のようになっています。それぞれは各16ビット幅で、0〜4Word、25〜47Wordはシステムの予約領域となっておりユーザーが操作することはできません。
5WordのControl Bitは、IPアドレスの変更・オートスキャンモードの有効/無効を設定するビットです。
EEPROMは何もデータがない場合、0xFFFFです。IPアドレスの変更を有効にした場合、ビット0が0になります。オートスキャンモードを有効にした場合、ビット2が0となります。

■EEPROMを操作するコマンド

EEPROMを操作するコマンドは5つあります。EEPROMからデータを読み込むReadコマンド、内容を消去するEraseコマンド、内容を書き込むWriteコマンド、EEPROMへの書き込みを有効にするWrite Enableコマンド、EEPROMへの書き込みを無効にするWrite Disableコマンドです。
コマンドを送信する場合には、コマンドに続きアドレス値、送信するデータの上位8ビット、続いて下位8ビットを送信します。(MSB先頭です)

実行される内容	EEPROMのデータを読み込む
コマンド(16進数)	'R (0x27 0x52) ※2バイト
引数	アドレス値0x05〜0x24 0x00 0x00 ※3バイト
戻り値	0x52に続き、アドレス値、MSB、LSB ※4バイト
詳細	指定したアドレス値のデータを読み込みます。 各アドレスは16ビットですので、戻り値は上位8ビット、下位8ビットの順番となります。
例	アドレス5のデータを読み込んだ場合 0x27 0x52 0x05 0x00 0x00 (送信) 0x52 0x05 0xFF 0x00 (受信) この2バイトが値

※VB6でWinsockコントロールを使用している場合には下記のようなプログラムになります。
Winsock1.SendData "'R" + Chr$(Address) + Chr$(0) + Chr$(0)

実行される内容	モジュールをリセットする ※EEPROMの内容を反映させる
コマンド(16進数)	'@ (0x27 0x40) ※2バイト
引数	0x00 0xAA 0x55 ※3バイト
詳細	ETH−50Bをリセットします。 すべてのポートは入力ポート設定又はEEPROMに設定された内容になります。 EEPROMの設定内容が反映されます。 このリセットを行った場合には、10ミリ秒以上経過してから動作を開始させるようにします。

実行される内容	EEPROMにデータを書き込む
コマンド(16進数)	'W (0x27 0x57) ※2バイト
引数	アドレス値0x05〜0x24 書き込む値の上位8ビット(MSB) 書き込み値の下位8ビット(LSB) ※3バイト
戻り値	なし
詳細	指定したアドレスにデータを書き込みます。 アドレス値を指定した後、書き込むデータの上位8ビットを、続けて下位8ビットを送信します。 なお、書き込みコマンド実行に際しては、このコマンド実行前に「書き込み有効コマンド('1)」を送信しておく必要があります。

実行される内容	EEPROMのデータを消去する
コマンド(16進数)	'E (0x27 0x45) ※2バイト
引数	アドレス値0x05〜0x24 0xAA 0x55 ※3バイト
戻り値	なし
詳細	指定したアドレスのデータを消去します。消去されると値はFFFFとなります。 消去コマンド実行に際しては、このコマンド実行前に「書き込み有効コマンド('1)」を送信しておく必要があります。

実行される内容	EEPROMへの書き込みを有効にする
コマンド(16進数)	'1 (0x27 0x31) ※2バイト
引数	0x00 0xAA 0x55 ※3バイト
戻り値	なし
詳細	EEPROMへの書き込みや消去などの変更を有効にするコマンドです。書き込みや消去コマンド実行前に、このコマンドを送信してEEPROMへの変更を許可します。

実行される内容	EEPROMへの書き込みを無効にする
コマンド(16進数)	'O　(0x27　0x30)　　※2バイト
引数	0x00 0x00 0x00 ※3バイト
戻り値	なし
詳細	EEPROMへの書き込みや消去などの変更を無効にするコマンドです。このコマンドが送信されると、EEPROMへの変更は一切できなくなります。

■EEPROMの値が変更できないようにロックする

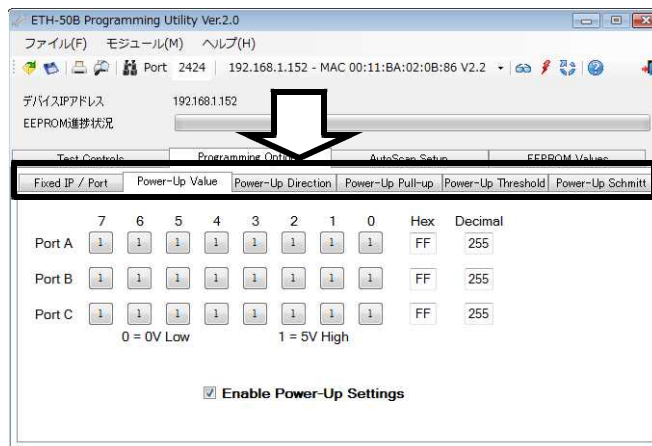
ジャンパー設定のJ1をショートすると、EEPROMへの書き換えができなくなります。J1をオープンにすると、自由にEEPROMへデータを書き込んだり、値を変更したりできます。

設定を変更したくない場合にはJ1をショートして設定変更をできなくすることをお奨めします。

パワーアップ時の初期設定

ETH-50Bでは、ポートの入出力方向やプルアップ設定などの各種設定をEEPROMに記録させておくことができます。EEPROMに記憶させておくと、ETH-50B電源投入時に初期状態として設定内容が自動的に反映されます。

■ユーティリティソフトウエアで設定を記憶させる

設定は、ETH-50Bのユーティリティソフトウエアにて行うことができます。Programmingタブへ移動すると各設定がタブ毎に分かれています。

1　設定したい項目のタブに切り替えます。
　設定は、Fixed IP(IPアドレスの設定)と、その他の設定とで分かれています。

2　各タブにおいて初期値(本体の電源投入時の状態)を設定します。

3　"Enable Power Up Settings"のチェックボックスにチェックを入れます。

4　EEPROM Valueタブに切り替えて、"EEPROMに書き込み"ボタンを押すと、設定内容がETH-50BのEEPROMに書き込まれます。
　エラーが表示される場合には、何度か繰り返し実行してください。

5　ETH-50Bをリセット又は再起動します。
　リセットは、メニューバーの"モジュール"→"モジュールの再起動"で行うことができます。

インターネット経由でETH-50Bを操作する（オプション）

※この項目はオプションです。設定に関するご質問などにはお答えできませんのであらかじめご了承ください。

ETH－50BにグローバルIPアドレスを割り振ればインターネット経由で各種操作が可能です。
グローバルIPアドレスは、プロバイダーなどから取得することができます。ETH－50Bはポート2424を使用しますので、ファイヤーウォールを通過する場合にはポート2424を解放します。

イーサネットなどのローカルエリアネットワークに接続されているETH－50Bをインターネット経由で操作するには、ルーターの設定が必要です。ルーター側で、ETH－50Bに割り当てられているプライベートIPアドレスとポート2424をWAN側から通過させられるよう、ポートフォワーディングの設定をする必要があります。
ポートフォワーディングはアドレス変換テーブルを静的に設定します。ポートフォワーディングを設定しておくと、WAN側に送信されたデータをLAN側の特定のIPアドレスとポートあてに転送します。

※この機能の名称はメーカーによって異なります。ローカルサーバ機能・静的NAT・SUAサーバ・アドレス変換テーブルの追加などと呼ばれることがあります。

ETH-50Bの仕様

■絶対定格

項　目	最大定格値
動作温度範囲	－65℃～+50℃
動作時の周囲環境温度	－40℃～+45℃
電源電圧	0V～+35V(DC)
DC入力電圧(信号線)	最大+5.6V
DC出力電流(各信号線)	最大45mA
DC出力電流(全体の総出力電圧)	最大210mA
入力ピンに入力できる最大DC入力電流	±500µA
消費電力	最大3W

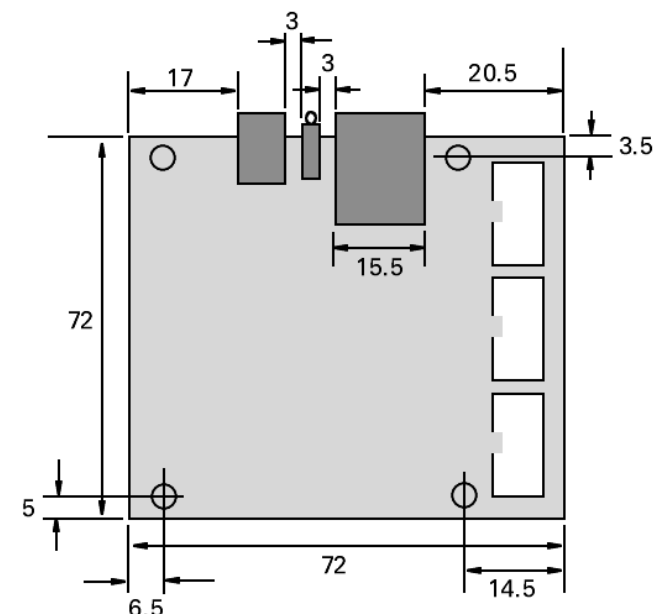
■電気的特性(周囲温度25℃、電源電圧24Vの場合)

項目	条件	最小	標準	最大
DC電源電圧		8V		35V
消費電力			1.1W	
温度範囲		0℃		45℃
湿度範囲		0%RH		85%RH
ロジック電圧(Low論理) TTL CMOS シュミットトリガー		 0V 0V 0V		 0.8V 1.5V 0.75V
ロジック電圧(High論理) TTL CMOS シュミットトリガー		 2.0V 3.5V 4.25V		 5.0V 5.0V 5.0V
入力漏れ電流		－3.0µA		+3.0µA
プルアップ電流		200µA	400µA	600µA
出力High電圧	負荷14mA時	4.3V		
出力Low電圧	負荷25mA時			0.6mA

※本製品は、周囲の環境温度が40℃よりも高くならない場所での使用をお願い致します。それよりも高くなる場合には、場合によってはCPU周辺に放熱板を付けるなどの熱対策が必要になる場合があります。

■その他の仕様

項　目	内容
搭載プロトコル	ARP, BOOTP, DHCP, ICMP, UDP/IP
データ通信	UDP/IP
物理層規格	10BASE－T
コネクタ	RJ－45コネクタ
CPU	SX48 (RISC)
MACアドレス	書き込み済み(Elexol社登録アドレス)
基板サイズ	72mm×72mm×24mm
IPアドレス設定	DHCP自動取得又は固定IPアドレス
生産国	オーストラリア Elexol Pty社製　開発元型式 Ether IO24R
供給形態	日本販売契約

■本体外寸

サポート情報

弊社では、ETH－50Bのサポートを行っております。以下のいずれかの方法でご質問をお寄せください。

■FAX番号　　03－3700－3548
■電子メール　　support@microtechnica.net

なお、他社製品に関することや自作回路に関するご質問にはお答え致しかねますのであらかじめご了承ください。

マイクロテクニカ
〒158－0094　東京都世田谷区玉川1－3－10
TEL: 03－3700－3535　FAX: 03－3700－3548